Procedia Computer Science
Volume 51, 2015, Pages 2474–2482
ICCS 2015 International Conference On Computational Science

Parallel solution of DDDAS variational inference problems

Vishwas Rao[1] and Adrian Sandu[1]

Virginia Tech., Blacksburg, Virginia, United States
visrao@vt.edu, asandu7@vt.edu

Abstract

Inference problems in dynamically data-driven application systems use physical measurements along with a physical model to estimate the parameters or state of a physical system. Developing parallel algorithms to solve inference problems can improve the process of estimating and predicting the physical state of a system. Solution to inference problems using the variational approach require multiple evaluations of the associated cost function and gradient, where the gradient is defined as the increase/decrease inflection point of the variable between two points. In this paper we present a scalable algorithm based on augmented Lagrangian approach to solve the variational inference problem. The augmented Lagrangian framework facilitates parallel cost function and gradient computations. We show that the methodology is highly scalable with increasing problem size by applying it for the Lorenz-96 model.

Keywords: DDDAS, time-parallel, variational inference

1 Introduction

Dynamically data-driven application systems (DDDAS [2, 4]) is a paradigm whereby simulations and measurements become a symbiotic feedback control system. An important application of DDDAS is the solution of inference problems where information from physical measurements is combined with a mathematical model to obtain estimates of the state or parameters of a physical system.

In this work, we develop a parallel framework to solve the inference problems. Our methodology is based on the variational framework and can have numerous applications in DDDAS. Examples include data assimilation for numerical weather prediction, the dynamic steering of the measurement process to improve overall forecasts [13], and dynamic location of faulty sensors.

Typically, there are two popular approaches to solve inference problems: variational and ensemble-based methods. The variational approach is based on optimal control theory, whereas the ensemble-based approaches is rooted in statistical estimation theory. In this paper, we focus on the variational approach. The variational approach requires the development of adjoint and tangent linear models. This method is popular in most national and international numerical weather forecast centers to provide the initial state for their forecast models. Variational

 Selection and peer-review under responsibility of the Scientific Programme Committee of ICCS 2015

doi:10.1016/j.procs.2015.05.356

approaches require applying an optimization algorithms, such as L-BFGS (Limited memory Broyden Fletcher Goldfard Shanno) [9] and conjugate gradient methods. The gradient is evaluated by solving the adjoint model. The entire process of evaluating the cost function, gradient, and subsequently using these computations in the optimization process is highly sequential.

The current generation of computers are becoming more and more parallel and the processor speed is not increasing rapidly. As our understanding of the physics improves, the computer model representing physics of the dynamical system becomes more and more complex. Hence it is necessary to develop parallel and scalable algorithms to solve inference problems.

Trémolet and Le Dimet have shown how variational data assimilation (4D-Var) can be used to couple models and to parallelize the computations in the spatial direction [14]. In [11], the author considers different data distribution strategies to perform scalable 4D-Var. However, the scalability is restricted to spatial parallelism. A scalable approach for variational data assimilation is presented in [6]. The authors achieve parallelism by dividing the 4D-Var into multiple 3D-Var sub-problems. In [1], the authors show how to apply augmented Lagrangian approach to constrained inference problems in the context of graphical models.

In this paper, we present a scalable parallel algorithm for solving the 4D-Var problem. The algorithm evaluates the important components of 4D-Var namely, cost function and gradient computations in parallel. The most important contribution of this work is that it achieves parallelism in temporal direction and can be conveniently extended to achieve parallelism in spatio-temporal directions. This approach is similar to our previous work on adjoint-based time-parallel integration [12]. The paper is organized as follows: Section 2 gives a brief description about data assimilation and 4D-Var. In Section 3 we show how the 4D-Var can be reformulated using augmented Lagrangian technique so that the gradient and cost function evaluations can be parallelized. Section 4 gives a detailed description of the parallel algorithm. Section 5 shows the numerical experiments with the chaotic Lorenz-96 model and Section 6 gives concluding remarks and future directions.

2 Data assimilation and 4D-Var

Data assimilation (DA) is the fusion of information from imperfect model predictions and noisy data, to obtain a consistent description of the state of a physical system [5, 7]. To obtain an estimate of the true state of a system $\mathbf{x}^{\mathrm{true}}$, three different sources of information are combined: the prior information, the model, and the observations. The best estimate that optimally fuses all these sources of information is called the analysis $\mathbf{x}^{\mathrm{a}}$. The following subsections briefly describe the sources of information.

Prior information. The prior information encapsulates our current knowledge of the system. The prior information is captured by the background estimate of the state $\mathbf{x}^{\mathrm{b}}$ and the corresponding background error covariance matrix $\mathbf{B}$.

The model. The model captures our knowledge about the physical laws that govern the evolution of the system. The model evolves an initial state $\mathbf{x}_0 \in \mathbb{R}^n$ at the initial time t_0 to future states $\mathbf{x}_i \in \mathbb{R}^n$ at future times t_i. A general model equation is represented as follows:

$$\mathbf{x}_i = \mathcal{M}_{0,\mathrm{i}}\left(\mathbf{x}_0\right) . \tag{1}$$

The observations. Observations represent the snapshots of reality available at discrete time instances. Specifically, measurements $\mathbf{y}_i \in \mathbb{R}^m$ of the physical state are taken at times t_i, $i = 1, \cdots, N$

$$\mathbf{y}_i = \mathcal{H}^{\mathrm{t}}\left(\mathbf{x}^{\mathrm{true}}(t_i)\right) - \varepsilon_i^{\mathrm{meas}}, \quad i = 1, \cdots, N.$$

The observation operator $\mathcal{H}^{\mathrm{t}}$ maps the physical state space onto the observation space. The measurement errors are denoted by $\varepsilon_i^{\mathrm{meas}}$.

The model state is related to observations by the following relation:

$$\begin{aligned} \mathbf{y}_i &= \mathcal{H}(\mathbf{x}_i) - \varepsilon_i^{\mathrm{obs}}, \quad i = 1, \cdots, N, \\ \varepsilon_i^{\mathrm{obs}} &= \varepsilon_i^{\mathrm{repres}} + \varepsilon_i^{\mathrm{meas}}. \end{aligned} \tag{2}$$

The observation operator $\mathcal{H}$ maps the model state space onto the observation space. The observation error term $\left(\varepsilon_i^{\mathrm{obs}}\right)$ accounts for both measurement and representativeness errors $(\varepsilon_i^{\mathrm{repres}})$. The measurement errors can be attributed to faulty equipments. The representativeness errors are due to the inaccuracies of the numerical approximation inherent to the model.

2.1 Four dimensional variational data assimilation (4D-Var)

In 4D-Var, one finds the control variable which minimizes the mismatch between the model forecasts and the observations. In strong-constraint 4D-Var, the control parameters are the initial conditions $\mathbf{x}_0$; they uniquely determine the state of the system at all future times via the model equation (1). The background state is the prior knowledge of the initial conditions $\mathbf{x}_0^{\mathrm{b}}$. Given the background value of the initial state $\mathbf{x}_0^{\mathrm{b}}$, the covariance of the initial background errors $\mathbf{B}_0$, the observations $\mathbf{y}_i$ at t_i and the corresponding observation error covariances $\mathbf{R}_i$, $i = 1, \cdots, N$, the 4D-Var problem provides the estimate $\mathbf{x}_0^{\mathrm{a}}$ of the true initial conditions by minimizing the following cost function:

$$\mathcal{J}(\mathbf{x}_0) = \frac{1}{2}\left(\mathbf{x}_0 - \mathbf{x}_0^{\mathrm{b}}\right)^T \mathbf{B}_0^{-1}\left(\mathbf{x}_0 - \mathbf{x}_0^{\mathrm{b}}\right) + \frac{1}{2}\sum_{i=1}^{N}\left(\mathcal{H}(\mathbf{x}_i) - \mathbf{y}_i\right)^T \mathbf{R}_i^{-1}\left(\mathcal{H}(\mathbf{x}_i) - \mathbf{y}_i\right), \tag{3}$$

subject to the constraint posed by the model equation (1). The 4D-Var can be formulated as

$$\begin{aligned} \mathbf{x}_0^{\mathrm{a}} = &\arg\min_{\mathbf{x}_0} \quad \mathcal{J}(\mathbf{x}, \mathbf{x}_0) \\ &\text{subject to} \quad (1). \end{aligned} \tag{4}$$

The optimal solution for the inverse problem (4) can be found by using the adjoint sensitivity approach described extensively in [8] and [16]. The solution procedure using the adjoint sensitivity procedure cannot be parallelized as it is. In order to parallelize the solution procedure, we reformulate the problem using the augmented Lagrangian framework.

3 4D-Var using the augmented Lagrangian framework

The 4D-Var cost function is given by

$$\mathcal{J} = \frac{1}{2}\left(\mathbf{x}_0 - \mathbf{x}_0^{\mathrm{b}}\right)^T \mathbf{B}_0^{-1}\left(\mathbf{x}_0 - \mathbf{x}_0^{\mathrm{b}}\right) + \frac{1}{2}\sum_{\mathrm{k}=1}^{\mathrm{N}}\left(\mathcal{H}(\mathbf{x}_{\mathrm{k}}) - \mathbf{y}_{\mathrm{k}}\right)^T \mathbf{R}_{\mathrm{k}}^{-1}\left(\mathcal{H}(\mathbf{x}_{\mathrm{k}}) - \mathbf{y}_{\mathrm{k}}\right). \tag{5}$$

The cost function (5) is subject to the following model constraints:

$$\mathbf{x}_{\mathrm{k+1}} = \mathcal{M}_{\mathrm{k,k+1}}\left(\mathbf{x}_{\mathrm{k}}\right), \quad \mathrm{k} = 0, 1, \cdots, N-1\,. \tag{6}$$

In equation (6) $\mathcal{M}_{\mathrm{k,k+1}}$ represents the model, given the physical state, $\mathbf{x}_{\mathrm{k}}$ at time t_{k} evaluates the state $\mathbf{x}_{\mathrm{k+1}}$ at $t_{\mathrm{k+1}}$. The augmented Lagrangian associated with (5) with (6) as the constraints is given by [15]:

$$\begin{aligned}
\mathcal{L} \;=\;& \frac{1}{2}\left(\mathbf{x}_0 - \mathbf{x}_0^{\mathrm{b}}\right)^T \mathbf{B}_0^{-1}\left(\mathbf{x}_0 - \mathbf{x}_0^{\mathrm{b}}\right) + \frac{1}{2}\sum_{\mathrm{k=1}}^{\mathrm{N}}\left(\mathcal{H}\left(\mathbf{x}_{\mathrm{k}}\right) - \mathbf{y}_{\mathrm{k}}\right)^T \mathbf{R}_{\mathrm{k}}^{-1}\left(\mathcal{H}\left(\mathbf{x}_{\mathrm{k}}\right) - \mathbf{y}_{\mathrm{k}}\right) \\
& - \sum_{\mathrm{k=0}}^{\mathrm{N-1}} \lambda_{\mathrm{k+1}}^T\left(\mathbf{x}_{\mathrm{k+1}} - \mathcal{M}_{\mathrm{k,k+1}}\left(\mathbf{x}_{\mathrm{k}}\right)\right) \\
& + \frac{\mu}{2}\sum_{\mathrm{k=0}}^{\mathrm{N-1}}\left(\mathbf{x}_{\mathrm{k+1}} - \mathcal{M}_{\mathrm{k,k+1}}\left(\mathbf{x}_{\mathrm{k}}\right)\right)^T \mathbf{P}_{\mathrm{k}}^{-1}\left(\mathbf{x}_{\mathrm{k+1}} - \mathcal{M}_{\mathrm{k,k+1}}\left(\mathbf{x}_{\mathrm{k}}\right)\right)\,,
\end{aligned} \tag{7}$$

where, λ are Lagrange parameters, $\mathbf{R}$ is the observation error covariance matrix, and $\mathbf{P}$ is a scaling matrix. The gradient of the augmented Lagrangian can be computed in parallel over the different sub-intervals.

4 The parallel algorithm

The solution procedure for 4D-Var requires multiple cost function and gradient evaluations. In this Section we describe a detailed procedure to evaluate the cost function and the gradients in parallel. Equation (7) can be used to carry out the cost function computations in parallel. Each cost function computation requires one model evaluation per processor. To complete the gradient evaluation, each processor needs to compute one forward and adjoint model. It should be noted that to begin the optimization, we need an initial guess for $\mathbf{x}$ and λ. The initial guess for $\mathbf{x}_{\mathrm{k}}$'s can be obtained by performing a forward integration using the background value for initial conditions. At optimality, we know that

$$\mathbf{B}_0^{-1}\left(\mathbf{x}_0 - \mathbf{x}_0^{\mathrm{b}}\right) + \lambda_0 = 0 \tag{8}$$

holds. Hence we can use (8) to obtain an initial guess for λ.

The optimization proceeds in two stages: inner and outer iterations. Inner iterations solve the optimization for a particular value of μ and λ. Subsequently, the values are μ and λ are updated after every inner iteration. The update after iteration k is carried out as follows:

$$\mu^{k+1} = \rho\mu^k \tag{9a}$$

$$\lambda_i^{k+1} = \lambda_i^k - \mu^k\left(\mathbf{x}_i - \mathcal{M}\left(\mathbf{x}_{i-1}\right)\right) \quad, i = 1, \ldots, N\,. \tag{9b}$$

Different strategies to update μ and λ can be used [3]. To achieve temporal parallelism, the assimilation window is divided into multiple sub-intervals. The state variable for the optimization process is the augmented variable containing the state vector at the beginning of each of the sub-interval. The initial state vector of the augmented Lagrangian in (7) is an augmentation of states at the beginning of every sub-interval obtained by propagating the background state using model equations. Figures 1 and 2 illustrate the convergence process. We start initially with a low value of μ, ensuring that the constraints are imposed loosely. Hence large discontinuities and large errors are seen in the green curve. Subsequently, μ is increased and as a result

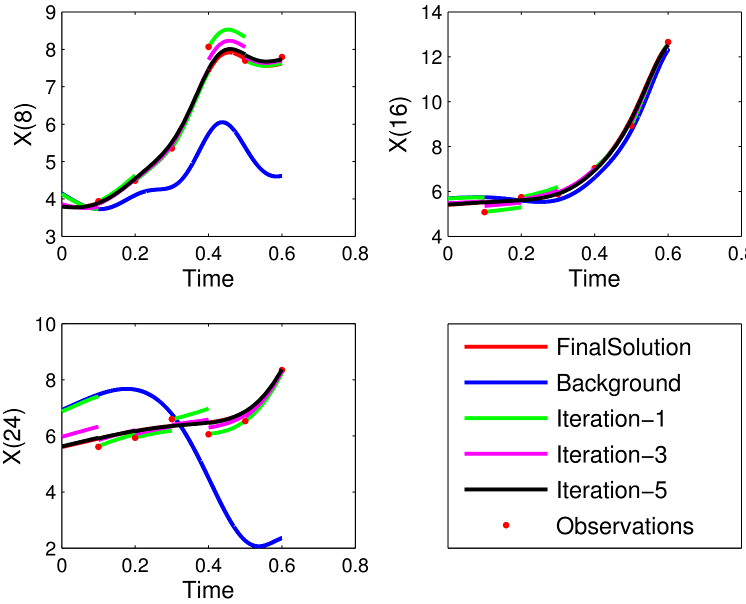


Figure 1: Convergence of the parallel algorithm towards the final solution. Three different variables are shown as an illustration.

the constraints are satisfied accurately. Black curve does not have any discontinuities, has very low errors compared to previous iterations and resembles the strong-constraint 4D-Var solution closely.

5 Numerical experiments

In this study we test the parallel implementation of the 4D-Var algorithm using the 40-variable Lorenz-96 model [10]:

$$\frac{\mathrm{d}\mathbf{x}_{\mathrm{i}}}{\mathrm{d}t} = \mathbf{x}_{\mathrm{i}-1}\left(\mathbf{x}_{\mathrm{i}+1} - \mathbf{x}_{\mathrm{i}-2}\right) - \mathbf{x}_{\mathrm{i}} + F\,, \tag{10}$$

where $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{40})^T \in \mathbb{R}^{40}$ is the state vector, and the forcing $F = 8$. We use synthetic observations to perform the data assimilation. The synthetic observations are generated by perturbing the reference trajectory with normal noise at uncertainty level of 5%. The background uncertainty is set to 8%. For convenience, no observations are taken at the beginning of an assimilation window. We compare the parallel and serial versions of the algorithm were run on a multi-core machine. The parallel algorithm was implemented in parallel-matlab.

5.1 Numerical results

We compare the performance of the proposed parallel implementation with that of the standard 4D-Var. We show both root mean square error (RMSE) and timing comparisons between the two implementations. The RMSE relative to the reference solution is used to compare the

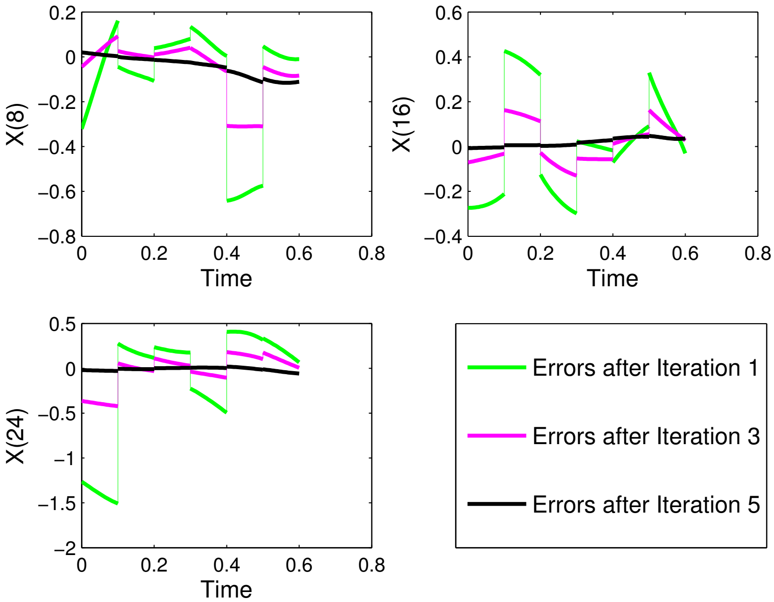
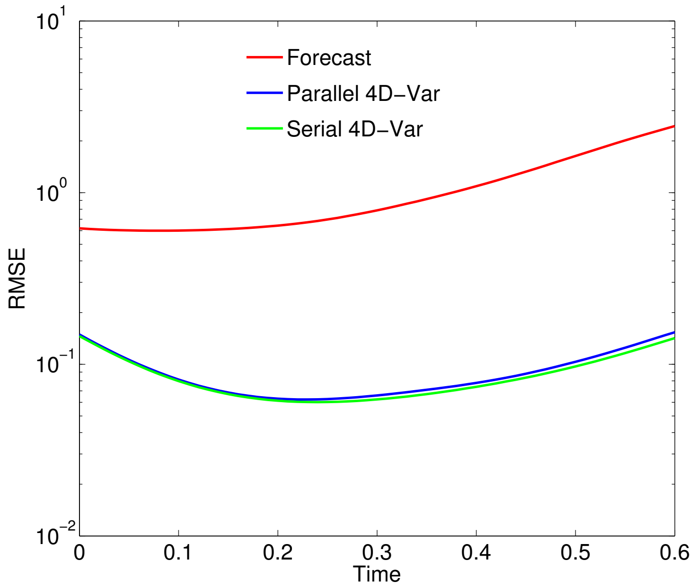


Figure 2: Error in the solution at different stages. The serial solution is taken as a reference to compute the errors. Three different variables are shown as an illustration.



(a) RMSE comparison for 6 Observations

Figure 3: RMSE comparison between serial and 4D-Var for Lorenz model.

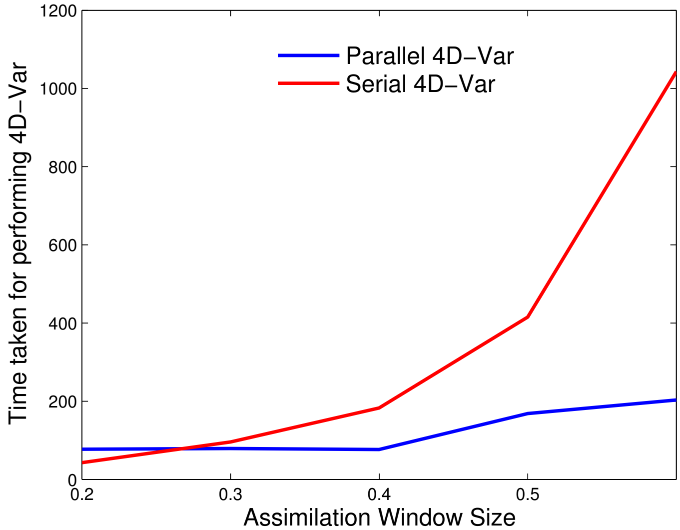


Figure 4: Timing Comparisons between serial and parallel 4D-Var (with 12 threads) for Lorenz model .

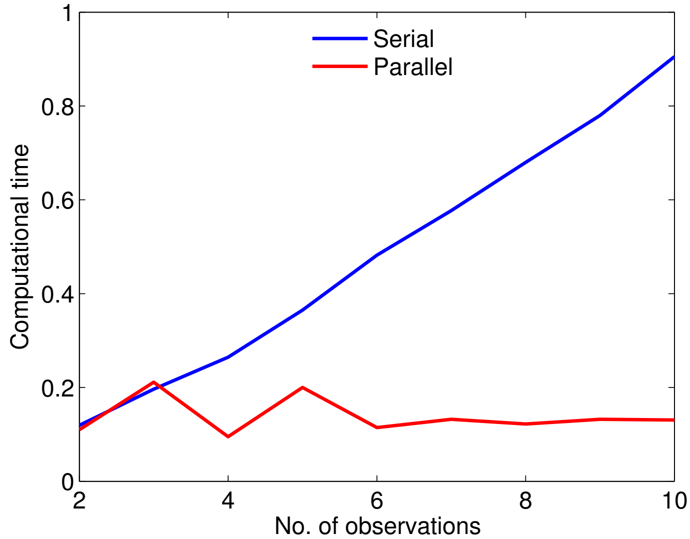
analysis obtained by both 4D-Var and parallel 4D-Var. The RMSE is given by

$$\mathbf{RMSE} = \sqrt{\frac{1}{\mathrm{N_{var}}}\sum_{i=1}^{\mathrm{N_{var}}}(\mathbf{x}_\mathrm{i} - \mathbf{x}_\mathrm{i}^{\mathrm{true}})^2} \quad , \tag{11}$$

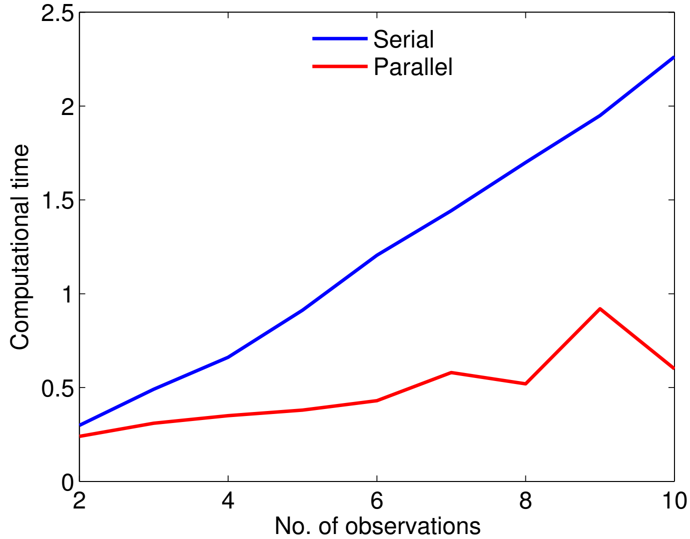
where $\mathbf{x}^{\mathrm{true}}$ is the reference solution, and $\mathbf{x}$ is the analysis (both propagated forward in time over the window). RMSEs for both implementations are calculated over the assimilation window by propagating the states, $\mathbf{x}$, and $\mathbf{x}^{\mathrm{true}}$ using the full model and applying (11). Figure 3(a) show the RMSEs of the parallel and the standard 4D-Var. We see that the parallel 4D-Var matches the serial version well. Figures 5(a) and 5(b) show the scalability of cost function and gradient computations. Figure 4 shows the timing comparisons of serial and parallel 4D-Var. It can be seen that the parallel algorithm is scalable in a weak sense, i.e., as the total computational work increases, the time taken to complete it is approximately constant.

6 Conclusions and future work

In this paper we have presented a scalable framework to solve variational DDDAS inference problems in parallel. Cost function computations and gradient evaluations, which are the main components of the algorithm, can be performed completely in parallel. We illustrate the proposed approach using a data assimilation test for the Lorenz-96 model. We see that the algorithm is feasible and gives real speed-up. The Lagrange parameter λ and μ can be updated to improve the convergence of the optimization. We intend to apply this algorithm to large scale problems such as, shallow water models and WRF (Weather Research and Forecast) models. A natural extension to this methodology is to include space-time parallelism. We intend to use certain ideas from [6] to further improve the scalability of the algorithm.



(a) Comparison for one cost function evaluation



(b) Comparison for one gradient evaluation

Figure 5: Timing comparisons for cost function and gradient computations between serial and parallel algorithms for Lorenz model.

Acknowledgements

This work was supported by AFOSR DDDAS program through the awards FA9550-12-1-0293-DEF and AFOSR 12-2640-06, managed by Dr. Frederica Darema.

References

[1] P. Aguiar, E. P. Xing, M. Figueiredo, N. A. Smith, and A. Martins. An augmented lagrangian approach to constrained map inference. In L. Getoor and T. Scheffer, editors, Proceedings of the 28th International Conference on Machine Learning (ICML-11), pages 169–176, New York, NY, USA, 2011. ACM.

[2] A. Aved, F. Darema, and E. Blasch. Dynamic data driven application systems. `www.1dddas.org`, 2014.

[3] E. G. Birgin and J. M. Martínez. Improving ultimate convergence of an augmented Lagrangian method. Optimization Methods and Software, 23(2):177–195, 2008.

[4] E. Blasch, G. Seetharaman, and K. Reinhardt. Dynamic data driven applications system concept for information fusion. Procedia Computer Science, 18(0):1999 – 2007, 2013. 2013 International Conference on Computational Science.

[5] R. Daley. Atmospheric data analysis, volume 2. Cambridge University Press, 1993.

[6] L. D'Amore, R. Arcucci, L. Carracciuolo, and A. Murli. A scalable approach for variational data assimilation. Journal of Scientific Computing, 61(2):239–257, 2014.

[7] E. Kalnay. Atmospheric modeling, data assimilation, and predictability. Cambridge University Press, 2003.

[8] S. Li and L. Petzold. Adjoint sensitivity analysis for time-dependent partial differential equations with adaptive mesh refinement. Journal of Computational Physics, Elsevier, 198:310–325, jul 2004.

[9] D. C. Liu and J. Nocedal. On the limited memory BFGS method for large scale optimization. Mathematical programming, 45(1-3):503–528, 1989.

[10] E. N. Lorenz. Predictabilty: A problem partly solved. In Proceedings of Seminar on Predictability, pages 40–58, 1996.

[11] J. Rantakokko. Strategies for parallel variational data assimilation. Parallel Computing, 23(13):2017 – 2039, 1997.

[12] V. Rao and A. Sandu. An adjoint-based scalable algorithm for time-parallel integration. Journal of Computational Science, 5(2):76 – 84, 2014. Empowering Science through Computing + BioInspired Computing.

[13] A. Sandu, A. Cioaca, and V. Rao. Dynamic sensor network configuration in infosymbiotic systems using model singular vectors. Procedia Computer Science, 18(0):1909 – 1918, 2013. 2013 International Conference on Computational Science.

[14] Y. Trémolet and F.-X. Le Dimet. Parallel algorithms for variational data assimilation and coupling models. Parallel Computing, 22(5):657 – 674, 1996.

[15] S. J. Wright and J. Nocedal. Numerical optimization, volume 2. Springer New York, 1999.

[16] S. Li Y. Cao and L. Petzold. Adjoint sensitivity analysis for differential-algebraic equations: algorithms and software. Journal of Computational and Applied Mathematics, Elsevier, 149:171–191, 2002.